Thomas Bennett - trb090020
CS 4395.001 - UTD Spring 2023
Chatbot Project

# Chatbot Report

## System Description

Have you ever participated in or witnessed a “quote chain” on an internet forum or comment thread? Someone in the thread quotes a song, a movie, a television show, or novel, or other well-known work of literature or other media. Someone else responds with the next line or sentence from the work, and thus the quote chain begins.

This project aims to create a chatbot that can do simple “quote chaining” with a specific work in mind: *Monty Python’s Flying Circus*. It is believed that the techniques used could create “quote chaining” for any work with sufficient written scripts or text. For example, The same techniques could be used for a novel, or a movie, provided the raw text of the work. The chatbot created for this project is able to reply to a prompt consisting of an exact quote from *Flying Circus*.

The chatbot uses a Python dictionary to create its behavior. The dictionary consists of key: value pairs where the key is a sentence from *Monty Python’s Flying Circus*, and the value is a Python Dictionary consisting of the following key: value pairs:

```
key:    {“this”: “[same as key]”
         “prev”: “[previous sentence]”
         “next”: “[next sentence]”}
```

Nesting a dictionary inside the value of a dictionary entry makes finding the next sentence very fast.

The knowledge base was scraped from a publicly available website [1] that seems to have the tacit approval or indifference of the BBC and the creators of *Monty Python’s Flying Circus* as to its existence. Unfortunately, there are numerous spelling errors and other transcription errors that sometimes cause the results to vary.

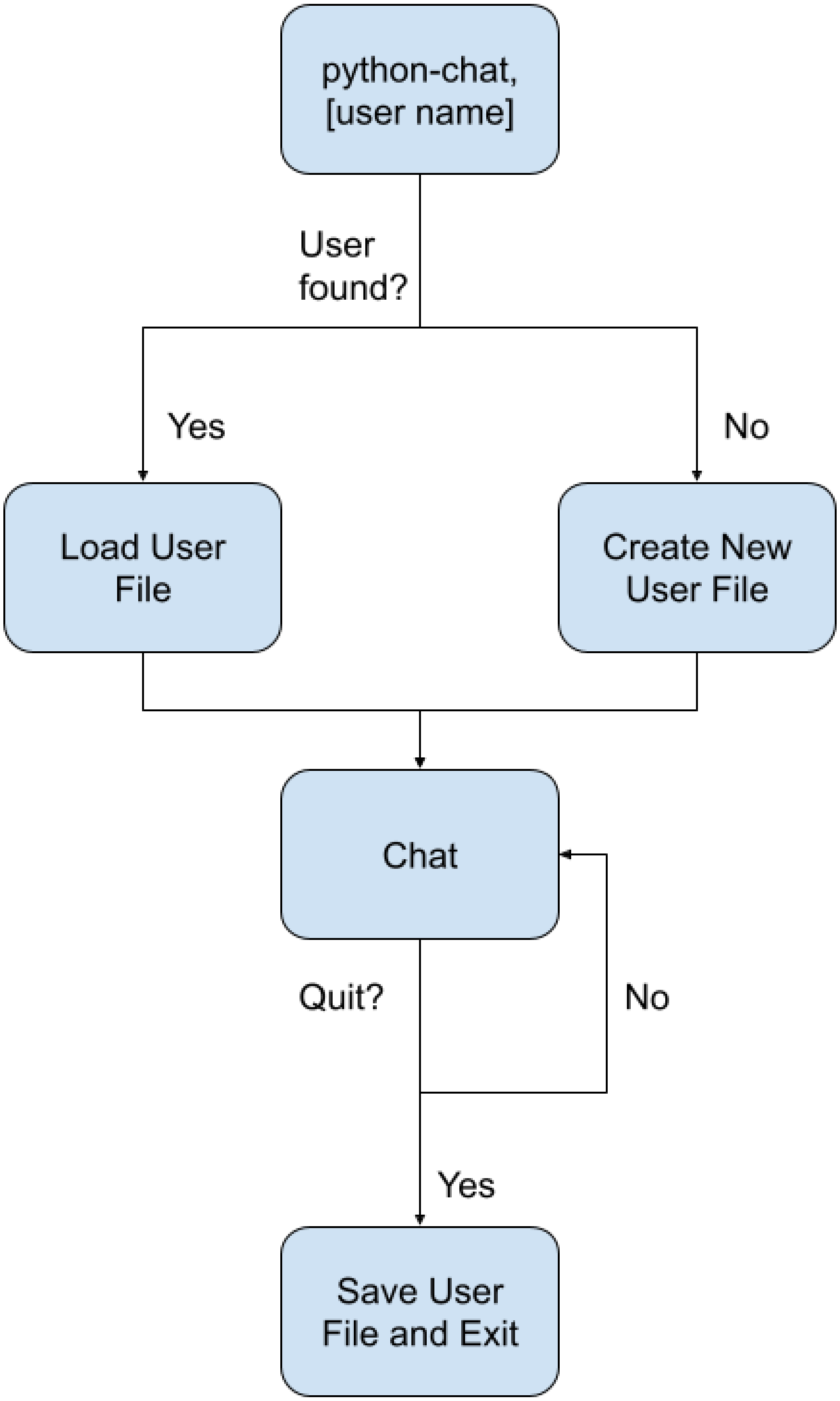
NLTK’s sent_tokenizer [2] function was used to tokenize the raw text into sentences. The sentences were then used to create the dictionary with the “this”, “prev”, and “next” data entries.

## How to use the bot

1. Run python-scraper.py
2. Run python-dict.py
3. Run python-chat.py with your name as an argument to the execution command

Thomas Bennett - trb090020
CS 4395.001 - UTD Spring 2023
Chatbot Project

# Diagram of Logic



Thomas Bennett - trb090020
CS 4395.001 - UTD Spring 2023
Chatbot Project

# Sample Dialogue

```
user>Next we have number four, 'Crunchy Frog'.
bot>'An, yes.'
user>Am I right in thinking there's a real frog in here?
bot>'Yes.'
user>What sort of frog?
bot>'A dead frog.'
user>Is it cooked?
bot>'No.'
user>What, a raw frog?
bot>('We use only the finest baby frogs, dew-picked and flown from Iraq,
cleansed in the finest quality spring water, lightly '
   'killed, and then sealed in a succulent Swiss quintuple smooth treble cream
milk chocolate envelope, and lovingly frosted '
   'with glucose.')
```

In the sample dialogue, a spelling error can be observed in the first response of the bot when it replies: “An, yes” to the first prompt. The bot provides the next sentence of the sketch based on what the user enters.

# Evaluation of the chatbot

## Strengths

The chatbot can very quickly look up the next sentence. The chatbot code, including the web scraper, and the script to form the knowledge base, is very simple and easy to understand.

## Weaknesses

The user has to provide an exact quote or the chatbot will not be able to find the sentence. Some of the sentences contain stage directions in parenthesis. The stage directions were left in since the source text contains them, and trying to figure out what the sentences would look like without them would require careful inspection of each individual sentence after being scraped from the source. This would be very time-consuming as there are 45 episodes, with hundreds of sentences each.

## Further Work

A function to do reverse quote chains could be easily implemented as the dictionary entry for each sentence already contains the previous sentence.

Further versions of the bot could use parsing to try to guess which sentence being prompted by the user is without being exactly the same as the key value.

Thomas Bennett - trb090020
CS 4395.001 - UTD Spring 2023
Chatbot Project

# Appendices

## Appendix A: Knowledge Base

Example entry in the knowledge base:

```
“Am I right in thinking there’s a real frog in here?”:
     {“this”: “Am I right in thinking there’s a real frog in here?”,
      “prev”: “An, yes.”,
      “next”: “Yes.”}
```

## Appendix B: User Models

User models consist of the user’s name and the most recent prompt provided to the bot saved in a Python list. The list is saved to a pickle file.

```
[‘thomas’, ‘Am I right in thinking there’s a real frog in here?’]
```

Thomas Bennett - trb090020
CS 4395.001 - UTD Spring 2023
Chatbot Project

# References

[1] Monty Python's Flying Circus - Just the words

[2] nltk.tokenize.sent_tokenize